Journal of Trends in Computer Science and Smart Technology (ISSN: 2582-4104)
www.irojournals.com/tcsst/

An Optimized Inter Planetary File System Framework Integrating Federated Learning and Blockchain to Bridge Interoperability and Latency Gaps in Electronic Health Record Systems

Mhaske Varsha Dattatraya[1], Ashok Kumar P M.[2], Jadhav Hema Keshav[3], Devika Veerkumar Mehta[4]

[1, 2]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur District, AP, India.
[1]SVPM's COE, Malegaon(Bk), Maharashtra, India.
[3]Department of AIDS, [4]Department of IT, Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati, Maharashtra, India.

Email: [1]varshamhaske13@gmail.com, [2]profpmashok@gmail.com, [3]hemakjadhav@gmail.com, [4]devika.tashu@gmail.com

Abstract

In the current era, Electronic Health Record (EHR) systems are widely adopted to store and manage patients' medical information in digital form, as they allow doctors and healthcare professionals to view a patient's complete medical information in an instant. The use of EHR makes healthcare faster, more accurate, and safer, and is therefore an important part of the future of digital healthcare. However, it faces many obstacles in terms of seamless integration (interoperability) and low-latency data acquisition, which directly impacts real-time medical decision-making and the quality of patient care. Integrating Blockchain Technology for EHR management with the InterPlanetary File System (IPFS) and federated learning can improve system performance by reducing the high latency of data retrieval, despite challenges like non-Independent and Identically Distributed (IID) data, client drift, and intermittent connectivity across hospital nodes. To address these challenges, we introduced Adaptive Contextual IPFS Retrieval (ACIR) and asynchronous aggregation. We tested our framework in a simulated environment representing 1,000 hospitals, and the results were promising. Data could be retrieved 65% faster, model training finished 38% sooner, and the system's overall performance improved by 42%. Most importantly, we achieved these improvements while maintaining full compliance with HIPAA and GDPR data privacy standards.

Keywords: Electronic Health Records (EHR), Interoperability, Data Retrieval Latency, IPFS, Blockchain, Distributed Ledger Technology (DLT), Hybrid Algorithm, Adaptive Contextual IPFS Retrieval (ACIR), FHIR, Decentralized Systems, Healthcare IT, Performance Optimization, Federated Learning (FL), Health Insurance Portability and Accountability Act (HIPAA), General Data Protection Regulation (GDPR).

DOI: https://doi.org/10.36548/jtcsst.2025.3.013
Received: 13.07.2025, received in revised form: 18.08.2025, accepted: 04.09.2025, published: 16.09.2025

1. Introduction

If we look at it today, digital transformation has brought about radical changes in the healthcare sector. Electronic health record (EHR) systems have become an integral part of healthcare. They have brought about significant and transformative change in how patient information is stored, used, and accessed.to make patient care more effective by reducing medical mistakes, and making administrative tasks easier and more efficient these systems digitally capture and integrate a patient's entire medical history including diagnoses, medications, lab tests, imaging reports, and doctor's notes [1],[2]. The vision is a future where comprehensive patient data is seamlessly available to authorized healthcare providers, fostering continuity of care, enabling personalized medicine, and driving evidence-based clinical decisions.

Although EHRs are very comprehensive and useful, there are still some challenges in EHR management that need to be addressed, such as interoperability and data retrieval latency. The current healthcare landscape features a fragmented ecosystem, where various electronic health record (EHR) systems, possibly proprietary and confined to separate organizations, face difficulties in effectively communicating with each other.

Even if Electronic Health Records (EHRs) are highly comprehensive and undeniably valuable, EHR management still faces important challenges particularly with interoperability and data retrieval delays. Today's healthcare environment is marked by a fragmented ecosystem in which multiple EHR systems are often proprietary and locked within individual organizations struggling to communicate seamlessly with one another. This lack of integration results in incomplete patient profiles, unnecessary repeat tests, and delays in delivering critical care, all of which compromise patient safety and reduce the overall effectiveness of treatment [1], [4]. Although initiatives such as Fast Healthcare Interoperability Resources (FHIR) and Health Level Seven (HL7) have made strides toward standardizing data exchange, inconsistent adoption and the complexity of integrating legacy systems continue to stand in the way. Therefore, actual "plug-and-play" interoperability, remains a challenging objective rather than a present-day reality [3], [5].

The main advantage of using blockchain technology is decentralization, immutability, and cryptographic security that enable transparent yet secure access control to sensitive patient information [6], [7], [1]. Maintaining a shared, tamper-proof record across multiple stakeholders, blockchain has the ability to ensure trust and enable more resilient, patient-centered approaches to healthcare data management. Storing large volumes of EHR data on the blockchain is often impractical due to limitations in scalability, transaction throughput, and high storage costs [8], [9]. IPFS offers off-chain storage of large or semi-static EHR files, such as medical images, genomic datasets, and detailed clinical reports., efficiently providing decentralized storage and content-based data retrieval.

Hybrid architectures that combine blockchain for metadata, access control, and content identifiers (CIDs) with IPFS for bulk data storage have demonstrated substantial potential in addressing the scalability limitations of purely blockchain-based solutions [10], [11], [12]. IPFS offers efficient decentralized storage and supports the large-scale distribution of data [13], [14]. Despite the promise of hybrid blockchain–IPFS models, a critical bottleneck remains: data retrieval latency. Blockchain offers secure storage and verification of the information retrieval process which involves the following steps:

- Generate the data hash from the ledger,

- Resolve it through the IPFS network to locate peers, and
- Finally, restore the encrypted record.

This system frequently results in unpredictable delays, which are particularly problematic in time-sensitive clinical settings, where even minimal latency in accessing patient information can jeopardize patient safety and treatment outcomes [15], [16], [17]. A key contributing factor is that current IPFS implementations were originally designed for generic file-sharing purposes and are not inherently optimized for the real-time, high-demand access patterns typical of EHR data. These systems were not built to handle the unique requirements of EHR data, which demand real-time responsiveness, frequent version updates, and support for highly diverse information types. Without mechanisms such as intelligent content routing, adaptive caching, and effective management of continuously evolving records, the performance of distributed healthcare environments remains inadequate [18], [19], [20].

This research directly addresses that gap by introducing an optimized IPFS framework tailored specifically for EHR systems. This study focuses on the issue of data retrieval latency in decentralized systems by preserving interoperability. We present a hybrid algorithm that combines intelligent mechanisms for content addressing, peer discovery, and data caching, specifically adapted to the IPFS framework. These mechanisms are explicitly designed to accommodate the diverse and continuously evolving characteristics of EHR datasets [21], [22]. Through this optimization of the retrieval pathway, the framework supports rapid, secure, and seamless access to patient information, thereby strengthening both the quality and timeliness of clinical decision-making [23], [24], [25].

Furthermore, this paper presents a hybrid Federated Learning (FL) integration with blockchain architecture, where locally trained models learn optimal strategies for data indexing, caching, and routing. With blockchain-managed metadata, the system can dynamically select the most efficient retrieval node using historical access patterns and ledger logs. This integration reduces latency and enhances the responsiveness and reliability of EHR systems [31], [32].

2. Related Work

The evolution of healthcare information systems has continuously pushed towards enhancing patient care through efficient data management and seamless information exchange. EHR systems are the cornerstone of this evolution, digitizing patient medical histories, diagnoses, medications, treatment plans, and more. While the widespread adoption of EHRs has undoubtedly brought numerous benefits, including improved accessibility to patient data, reduced medical errors, and enhanced administrative efficiency, their full potential remains hampered by significant challenges, primarily around interoperability and data retrieval latency. This section critically reviews the literature on EHR systems, focusing on various approaches to address these challenges and identifying the persistent gaps that necessitate novel solutions.

Each healthcare organization often deploys its own EHR system, leading to a landscape of isolated data silos. This fragmentation means a patient's complete medical history is rarely available in one place, requiring manual intervention, faxing, or arduous data entry to consolidate information when a patient moves between providers or specialists. Studies

document how this siloed approach leads to redundant tests, delayed diagnoses, and incomplete patient profiles [1], [2].

While centralized systems often employ robust security measures, they represent a single point of failure. A breach in one system can expose a vast amount of sensitive patient data, leading to severe privacy violations and compliance issues (e.g., HIPAA, GDPR). Furthermore, the lack of granular patient control over their data in these systems is a growing ethical concern [8], [9]. In a dynamic, large-scale network, identifying peers that hold the required data can introduce unpredictable latency, especially when data is dispersed or when peers are offline [17], [20]. The research will consider heterogeneous EHR data types, including structured clinical data (e.g., patient demographics, diagnoses, medication lists), semi-structured data (e.g., clinical notes, discharge summaries), and bulky unstructured data (e.g., medical images like X-rays, MRIs, and potentially large genomic files). [11], [1], [26]. The proposed optimizations will be designed to handle the unique retrieval requirements of each data category.

Tiwari and Kumar [27] present a blockchain-enabled IPFS-based architecture specifically tailored for secure healthcare data management, highlighting the benefits of decentralized storage and immutable access logs. Despite its advantages, storing large volumes of granular patient data directly on a blockchain is often impractical due to scalability limitations (transaction throughput) and high storage costs. While decentralized technologies have laid a foundation for secure and interoperable EHRs, the fundamental challenge of achieving sub-second retrieval latency for dynamic and diverse patient data within a fully distributed, IPFS-backed EHR system remains unaddressed and unoptimized [28].

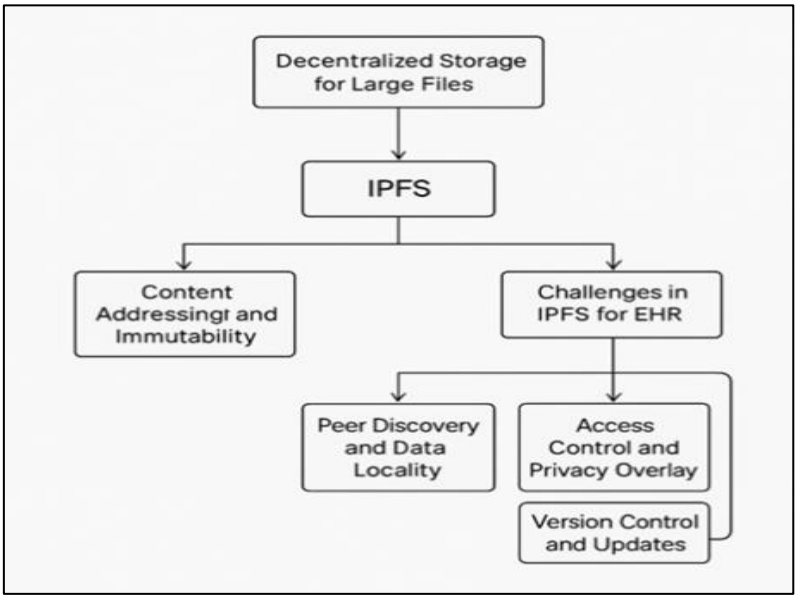

Figure 1. IPFS Challenges

The successful implementation and rigorous evaluation of the Optimized IPFS Framework for EHR Systems necessitate the use of a representative, diverse, and secure dataset. Due to the sensitive nature of real-world Electronic Health Records (EHR) and stringent privacy regulations (e.g., HIPAA, GDPR), directly accessing identifiable patient records for experimental purposes is ethically and legally restricted [13], [29]. Mabina and Mbotho [30] propose a hybrid security framework for 5G-enabled healthcare systems, focusing on enhancing encryption and access control in real-time data environments. While the approach

addresses core security concerns, it lacks empirical performance validation and does not explore interoperability with existing EHR standards or integration with privacy-preserving learning techniques.

Figure 1 provides an overview of using IPFS (InterPlanetary File System) for the decentralized storage of large files, specifically in the context of Electronic Health Records (EHRs). The central block represents IPFS as the core storage layer. On the left, it highlights Content Addressing and Immutability, supported by Peer Discovery and Data Locality essential features of IPFS. On the right, it outlines Challenges in IPFS for EHR, focusing on Access Control and Privacy Overlay, as well as Version Control and Updates. In essence, the diagram distinguishes between the native strengths of IPFS and the specific challenges it faces when applied to sensitive healthcare data.

This critical review highlights that while significant strides have been made in securing and distributing EHR data, a crucial bottleneck remains: the real-time, low-latency retrieval of comprehensive patient information in a seamlessly interoperable manner, particularly within distributed environments like those leveraging IPFS. The existing hybrid models, while addressing scalability, have not fully optimized the data retrieval pathway to meet the demanding performance requirements of modern healthcare.

3. Proposed Work

The proposed method illustrates a federated learning and blockchain-integrated architecture for secure and low-latency EHR management. Figure 2 provides the overall architecture of the proposed system.

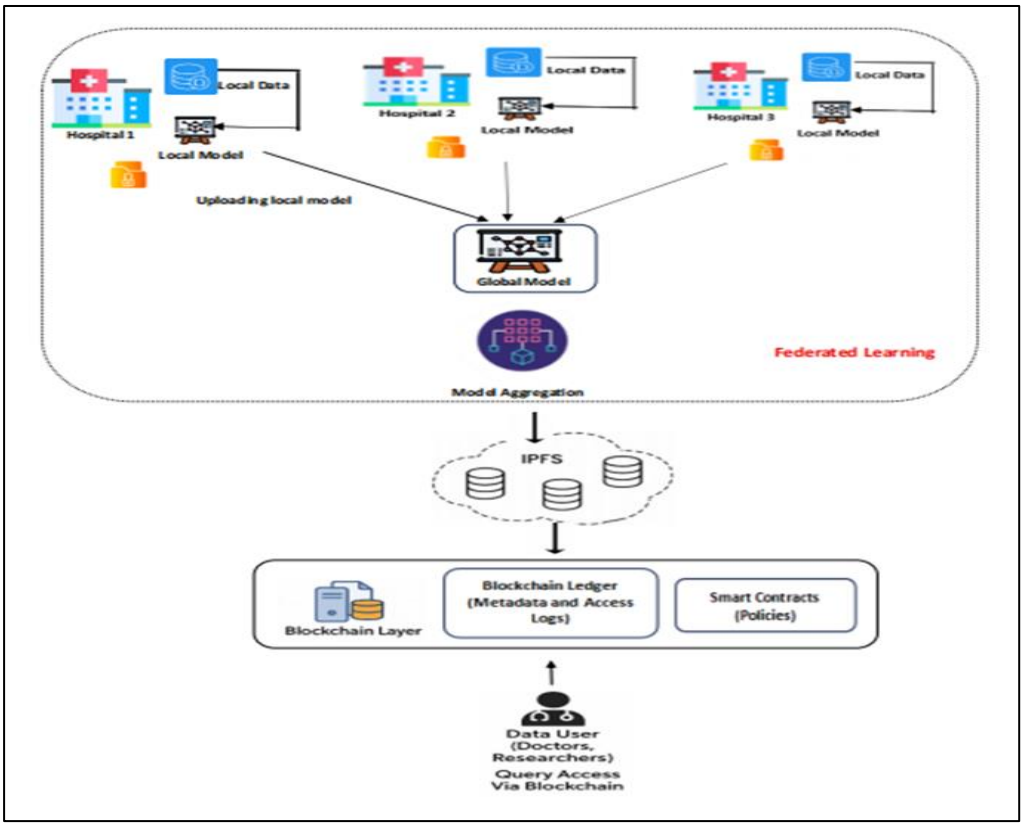

Figure 2. The Architecture of Proposed System

This system architecture integrates Federated Learning, Blockchain Technology, and optimized IPFS to facilitate secure and scalable EHRs data management across a network of hospitals. Each hospital trains its machine learning models locally on confidential data and sends only encrypted model updates to the blockchain. Smart contracts enforce patient consent, manage version history, and control data access. Through asynchronous aggregation, model updates are incorporated without needing simultaneous participation from all nodes. Both EHRs and model snapshots are stored off-chain using IPFS, with CIDs recorded on the blockchain. The system applies Bio-Keyed AES-256 encryption, where encryption keys are partially derived from patient fingerprints in order to meet HIPAA and GDPR standards, . In real-world use, hospitals upload encrypted data and model updates, while authorized users such as researchers access the data securely.

3.1 Healthcare Providers

Healthcare Providers represent individual entities such as hospitals, clinics, or research institutions participating in the federated learning and blockchain-based EHR management framework. Each provider node consists of the following components:

(a) Local EHR Database: Maintains sensitive patient health data, including diagnostic records, prescriptions and treatment plans, laboratory test results, imaging reports, clinical notes, and visit histories.

(b) Federated Learning Node (Training): Each provider runs a local machine learning (ML) training node that learns from local data without uploading raw records to a central server. It builds predictive models (e.g., access frequency, urgency classification) and periodically shares model parameters (not data) with a central aggregator. Here, we propose a modified Federated Averaging (FedAvg) framework that enhances traditional FL for EHR. In traditional FedAvg, hospitals train locally and send their model updates to a server, which then aggregates all updates to create a global model. However, EHR data is sensitive, so there are risks of issues like data leakage. Additionally, hospital data is different (i.e., non-IID), so the global model may not perform well, and communication takes time, which slows down training. By considering these issues, we can improve FedAvg as follows:

I. Hospitals locally train on their own EHR data; instead of sending original updates, they use differential privacy (DP), in which they add noise to updates so raw data cannot be easily guessed. Additionally, by using homomorphic encryption, updates are encrypted before being sent to protect sensitive data.

II. Instead of applying simple aggregation, the server assigns greater weight to hospitals that share similar patient populations or contribute higher quality data. This guarantees that the global model better reflects the most relevant and reliable sources, resulting in improved performance across diverse healthcare settings.

III. Instead of sending updates at fixed intervals, updates are transmitted only when significant improvements occur, thereby reducing communication overhead and accelerating convergence. Clients calculate a local utility signal after training; if the utility gain is below a threshold, they skip sending large backbone updates and send only small adapters.

IV. Finally, to enhance clinical trust and interpretability, each participating hospital supplements its model updates with lightweight explainability artifacts, enabling healthcare professionals to trust the model.

(c) Local Data Storage: Stores all EHR data on-premise, enabling direct model training on-site. It supports internal access and caching mechanisms, and it reduces external transmission risks.

3.2 Off-Chain Storage

In this layer, encrypted and anonymized EHRs are stored using decentralized storage infrastructure, namely the InterPlanetary File System (IPFS). This separates clinically sensitive information from blockchain storage, which addresses the scalability and efficiency of data storage.

3.3 Blockchain Layer

The Blockchain Layer provides security that functions as the trust, access control, and audit backbone of the decentralized EHR system. Patient records are not directly stored on the blockchain, but it manages and secures access to off-chain data through immutable metadata, access control through smart contracts, and cryptographic logs that are maintained.

3.4 Data User

The Data users layer includes all authorized Users or entities which require access to EHRs for clinical, diagnostic, research, or audit purposes. This layer interacts with the blockchain and IPFS-based system. All users identity are verified through digital certificates or federated identity systems. A list of authorised users is given below:

a) Doctors can access patient records for diagnosis, treatment, and follow-ups.

b) Researchers can analyze anonymized data for public health studies, epidemiology, or AI training.

c) Auditors/Regulators verify compliance with medical data handling standards.

d) Patients view their own EHRs under self-sovereign identity

Flow of Execution is as Follow

1. Users submit a request to access specific EHR data (e.g., lab results, imaging, medical history).
2. The system uses smart contracts to check identity and credentials. It vrifies patient consent and access policies.
3. Smart contracts validate query conditions (e.g., data type, purpose, frequency).
4. If the request is approved, the smart contract returns the CID (Content Identifier) from the blockchain.

5. It enables retrieval of the encrypted data from IPFS and manages logs for the access event on-chain for traceability. Access decisions are transparent, policy driven, and tamper-proof.

3.5 Global Federated Aggregator

The Global Federated Aggregator serves as the central coordinating hub in the federated learning architecture. It serves as the central (but not data-storing) entity that collects and consolidates model updates from distributed hospital or clinic nodes to generate an improved global model without requiring access to raw patient data.

3.6 Hybrid Algorithm Adaptive Contextual IPFS Retrieval (ACIR)

This algorithm is carefully crafted to overcome the data retrieval delays commonly seen in existing IPFS-blockchain EHR systems. It does so by smartly optimizing content placement, peer discovery, and caching strategies. Figure 3 presents the overall workflow of the Hybrid ACIR algorithm, which blends traditional data management principles with decentralized network dynamics to deliver faster, more reliable access tailored to the diverse and time-sensitive needs of EHR data [6], [20]. In the Optimized IPFS Framework, the ACIR algorithm strengthens federated learning by reducing client drift and effectively handling non-IID data across diverse hospital environments. It groups hospitals with similar data into clusters, which allows for more focused and efficient model training. To improve performance, ACIR uses context-aware caching and adaptive content placement, reducing retrieval delays while ensuring that models remain relevant to local needs. This approach gives hospitals quick access to personalized or cluster-specific models, supports local fine-tuning, and promotes stable training by aligning updates with the data patterns of each site. When combined with blockchain-based access control, ACIR also ensures secure auditing and trust.

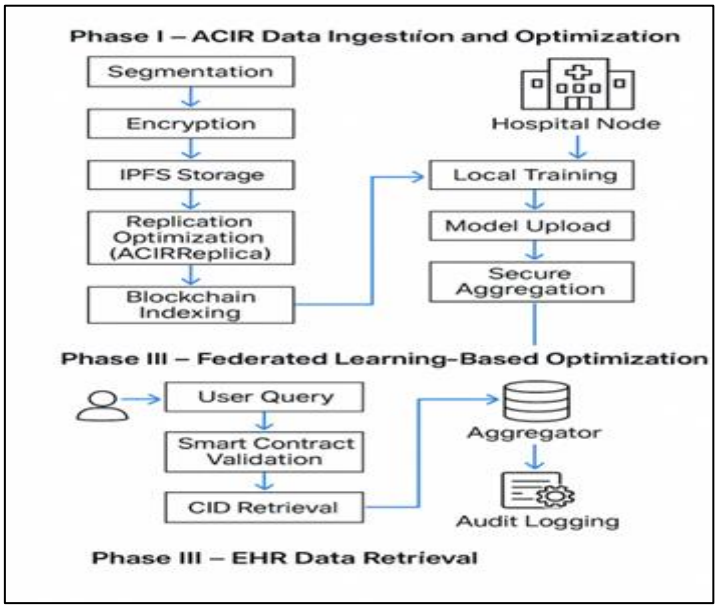

Figure 3. Hybrid Algorithm Adaptive Contextual IPFS Retrieval (ACIR)

The ACIR algorithm operates in two primary phases: the first is Data Ingestion and Optimization (when data is added to the system) and the second phase is Data Retrieval and Acceleration (when data is requested by a client).

Phase 1: Data Ingestion and Optimization

In this phase, patient EHRs undergo preprocessing, encryption, distributed storage, and indexing using the ACIR hybrid algorithm.

a) Segmentation

Raw EHR data (Draw) collected from each patient Pj is first converted into FHIR-compliant resources (r_1, r_2, ..., r_m). These resources are then broken down into smaller, fine-grained data objects (d_k), such as individual lab results, prescriptions, and diagnostic reports. By representing the data in such a modular way, the system supports incremental updates and enables selective retrieval of specific data elements.

Each patient's raw FHIR-compliant record $D_{raw}^{(Pj)}$ is parsed and segmented into discrete units:

$$D_k^{(Pj)} = \left\{ d_k | d_k \subset D_{raw}^{(Pj)} \right\}, \forall d_k \in FHIR_{types} \quad (1)$$

Where:

$D_{raw}^{(Pj)}$: Raw FHIR record for patient Pj

d_k : Fine-grained data object (e.g., Observation, LabResult, Imaging)

$D_k^{(Pj)}$: Set of segmented data objects

b) Encryption

Once the data objects d_k are segmented, each unit is anonymized to remove personally identifiable information (PII). After this, a unique symmetric key K_k is generated for every data object. This key is then used to encrypt the data using AES-256-GCM, which provides both strong confidentiality and integrity. By doing so, even though the encrypted objects are stored on the public IPFS network, their security is preserved. The management and distribution of encryption keys are efficiently handled through blockchain-based smart contracts, ensuring secure and transparent access control.

Each unit d_k is encrypted using AES-256 GCM with a symmetric key K_k:

$$d_k^{enc} == Enc_{AES\text{-}256}(d_k, K_k) \quad (2)$$

Where:

d_k^{enc} : Encrypted data object

$Enc_{AES\text{-}256}$: AES-GCM encryption function

K_k : Unique symmetric key for d_k

c) IPFS Storage

After encryption each data objects ($_{dk}E$) is assigned a content-addressed using a cryptographic hash to generate CIDs (Content Identifiers). This CID serves as a unique fingerprint for the data, ensuring that it can always be located and verified without duplication or tampering. The encrypted objects are then uploaded to the local IPFS node, making them available in a decentralized and distributed network. This approach not only guarantees secure storage but also provides built-in verifiability and resilience, since data can be retrieved from multiple peers in the network rather than relying on a single server.

Each encrypted data unit is uploaded to IPFS and assigned a content identifier (CID):

$$C_k = Hash(d_k^{enc}) \quad (3)$$

Where:

C_k: Content Identifier used by IPFS

Hash(.): Cryptographic hash function (e.g., SHA-256)

d) Replication Optimization (ACIRReplica)

The algorithm calculates a replication score R for each object and candidate IPFS peer node based on:

Paccess($_{dk}E$): Likelihood of being queried.

Uclinical($_{dk}E$): Clinical urgency or criticality.

Dnetwork(N_i): Network distance or latency between the requester and the node.

Top-k peers with the highest scores are selected for proactive replication, ensuring fast and reliable retrieval from geographically and logically optimal nodes.

Replication is determined by a score that considers access probability, clinical urgency, and network proximity:

$$R(d_k^{enc}, N_i) = \alpha \cdot P_{access}(d_k) + \beta \cdot U_{clinical}(d_k) - \gamma \cdot D_{network}(N_i, L_{client}) \quad (4)$$

Where:

$R(d_k^{enc}, N_i)$: Replication score for data d_k on node N_i

P_{access}: Predicted access frequency (e.g., via ML)

$U_{clinical}$: Clinical urgency score (predefined by data type)

$D_{network}$: Normalized latency between node and client

$\alpha + \beta + \gamma = 1$: Tuning coefficients

L_{client}: Location of anticipated data requester

Pin the encrypted data d_k^{enc} to the top-k_r nodes with the highest replication scores:

$$\text{ReplicaSet}(d_k^{enc}) = \text{Top-k}_{r,N_i}[R(d_k^{enc}, N_i)] \quad (5)$$

e) Blockchain Indexing

Each data object's CID, metadata, and encrypted key are stored in a blockchain transaction via smart contracts. A root CID pointing to a DAG structure in IPFS links all related CIDs, enabling efficient navigation and versioning. The blockchain guarantees immutability, traceability, and fine-grained access control for all EHR segments.

All metadata is registered immutably on the blockchain:

$$BC_{indexk}^{(Pj)} = \{(C_k, Meta_k, Ref(K_k)) | \forall d_k \in d_k^{(P_j)}\} \quad (6)$$

Where:

$Meta_k$: Metadata for each data chunk (type, version, timestamp)

$Ref(K_k)$: Reference or encrypted storage for key K_k

$BC_{indexk}^{(Pj)}$: Patient-specific blockchain record

Update the root CID (IPFS DAG root) for the latest patient record version:

$$RootCID_{new}^{(Pj)} = CID_{DAG}(\{C_k\}) \quad (7)$$

Stored using

$$SmartContract.Update((P_j, RootCID_{new}^{(Pj)}) \quad (8)$$

Phase II – Federated Learning-Based Optimization

This phase enables privacy-preserving AI model training across distributed hospital nodes, without transferring raw patient data.

a) Local Training at Hospital Node

Each provider uses local EHR data to train a machine learning model (e.g., predicting patient access patterns or optimizing CID caching strategies). This ensures that raw data never leaves the provider, preserving compliance with HIPAA/GDPR. Bio-Keyed Adaptive AES-256 (BKA-AES256) is a sophisticated encryption method that integrates fingerprint-based biometric authentication with AES-256 to protect Electronic Health Records (EHRs) in blockchain-enabled environments. A segment of the encryption key is generated from the user's fingerprint, binding the security process directly to their identity. Additionally, the algorithm employs a dynamic, session-specific S-Box to enhance encryption strength. While the encrypted EHRs are stored off-chain on optimized IPFS, access rights are managed through smart contracts on the blockchain. Access is permitted only when a valid fingerprint and user consent are provided.

b) Model Upload

The locally trained model parameters are shared with a central aggregator (not the data). All communication is secure and privacy-preserving, potentially using homomorphic encryption or differential privacy.

Each hospital node $i \in \{1,2,...,N\}$ trains a local model $w_i^{(t)}$ at round t using its local data D_i:

$$w_i^{(t+1)} = w_i^{(t)} - \eta \cdot \nabla L(w_i^{(t)}, D_I) \quad (9)$$

Where: η is the learning rate

∇L is the gradient of the local loss function L

D_I is the private dataset held by node i

c) Secure Aggregation

The aggregator performs federated averaging (FedAvg) or similar aggregation techniques to compute a global model.

Encrypted or differentially-private parameters are transmitted securely:

$$\text{Send}(\tilde{w}_i^{(t+1)} \text{ or } \hat{w}_i^{(t+1)}) \rightarrow \text{Aggregator} \quad (10)$$

Where:

All communication is over secure channels (e.g., TLS, SSL)

Aggregator cannot reconstruct local data from $\tilde{w}_i$ or $\hat{w}_i$

This model is sent back to all participating nodes, improving local intelligence on data access prediction, replication decisions, or routing paths.

Phase III – EHR Data Retrieval via Blockchain

This phase handles real-time user queries, policy validation, and low-latency content access.

a) User Query: Authorized users (e.g., doctors, researchers) initiate a request to retrieve patient data.

Let,

U_i: Authorized user i (e.g., a doctor or researcher)

Q_i: Query issued by U_i_

P_j: Patient j

R_j: Requested EHR resource of patient P_j

$A(U_i,R_j)$: Access function that returns 1 if U_i is authorized to access R_j , else 0.

Then the user query is granted only if:

$$A(U_i, R_j) = \begin{cases} 1, & \text{if } U_i \text{ satisfies access policy for } R_j \\ 0, & \text{Otherwise} \end{cases} \quad (11)$$

And the access request is modeled as:

$$Q_I(R_J) = \begin{cases} \text{Retrieve } R_j, & \text{if } A(U_i, R_j) = 1 \\ \text{Deny Request}, & \text{if } A(U_i, R_j) = 0 \end{cases} \quad (12)$$

This models the initial authorization check before EHR content identifier (CID) retrieval and is enforced by smart contracts on the blockchain layer.

b) Smart Contract Validation

The request is validated on the blockchain using pre-defined smart contracts. Validation criteria may include user role, patient consent, request frequency, and emergency override status.

Let:

U_i: Requesting user (e.g., doctor, researcher)

R_j: Requested EHR resource of patient Pj

$V(U_i,R_j)$: Smart contract validation function (returns 1 if valid, else 0)

Define:

$\rho(U_i)$: Role of user U_i

$\sigma(P_j,U_i)$: Consent flag (1 if patient P_j has consented to U_i, else 0)

$f(U_i,R_j)$: Frequency of past access requests by U_i for R_j

$\theta(U_i,P_j)$: Emergency override flag (1 if true emergency access, else 0)

Let f_{max} be the maximum allowed query frequency.

Then the validation function is:

$$V(U_I,R_J) = \begin{cases} 1, & \text{if } [\rho(U_i) \in Rallowed] \wedge [\sigma(P_j,U_i) = 1 \vee \theta(U_i,P_j) = 1] \wedge [f(U_i,P_j)) \leq fmax \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

$CID_{j,k}$: Content Identifier for the k-th encrypted data object of patient P_j

$CID_j = \{CID_{j,1}, CID_{j,2},\ldots,CID_{j,n}\}$: Full set of patient P_j's IPFS data objects Let $\phi(U_i,R_j) \subseteq CID_j$ be the subset of CIDs relevant to the requested record R_j.

Then the CID retrieval function $C(U_i,R_j)$ is defined as:

$$C(U_i,R_i) = \begin{cases} \phi(U_i,R_j), & if\ V(U_i,R_j) = 1 \\ \emptyset, & \text{if } V(U_i,R_j) = 0 \end{cases} \quad (14)$$

If validation passes (V=1 the smart contract returns the precise CIDs relevant to the request.

If validation fails, access is denied and no CIDs are disclosed.

The function $\phi(U_i,R_j)$ performs CID filtering, i.e., only those CIDs related to the specific data request R_j are returned, not the entire HER.

c) CID Retrieval

If approved, the smart contract reveals the CID(s) corresponding to the requested data. This enables precise access to encrypted EHR segments stored on IPFS.

Let: $V(U_I, R_J)$: Validation function from the smart contract (as defined earlier), where:

$$V(U_I, R_J) = \begin{cases} 1 & \text{if access is granted} \\ 0 & \text{Otherwise} \end{cases} \quad (15)$$

d) Data Access and Audit Logging

The encrypted content is retrieved from the nearest or most responsive IPFS peer based on the ACIR optimization.

Let:

$CID_{j,k}$: Content Identifier for the kth encrypted data object of patient P_j

$N=\{N_1, N_2,...,N_m\}$: Set of available IPFS nodes

$R(N_l,CID_{j,k})$: Replication score of $CID_{j,k}$ on node N_l, computed using:

$$R(N_l,CID_{j,k}) = \alpha P_{access}(CID_{j,k})+\beta.U_{clinical}(CID_{j,k})-\gamma.D_{latency}(N_l) \quad (16)$$

Where:

P_{access}: Probability of access (from FL access patterns)

$U_{clinical}$: Clinical urgency score

$D_{latency}(N_l)$: Network distance (latency) between node N_l and user location

α,β,γ: Weighting coefficients, $\alpha+\beta+\gamma=1$

e) Optimal Node Selection

The data is retrieved from:

$$N^* = arg\ max\ _{N_l \epsilon N} R(N_l, CID_{j,k}) \quad (17)$$

Then the user retrieves:

$$Data^E_{j,k} \leftarrow GET_{IPFS}(N^*, CID_{j,k}) \quad (18)$$

f) Audit Logging

Every access is logged on the blockchain via:

$$L(U_{i,} CID_{j,k\ ,} t, N^*) = log_{chain}(U_{i,} CID_{j,k\ ,} \text{timestamp t, Node } N^*) \quad (19)$$

Where:

L: Smart contract that records access

t: Timestamp of access

Here, the proposed method illustrates that Node N∗ is selected using ACIR's scoring model, and access is logged immutably on the blockchain using smart contracts. In this way, this mechanism ensures low-latency retrieval and verifiable auditability. All access events are logged immutably on the blockchain, providing an auditable trail for regulatory compliance and trust.

4. Results and Discussion

The system proposed can offer a solution for managing EHR in 100 hospitals. The models are locally trained inside each hospital on heterogeneous, non-IID data and are then projected onto clusters to enhance learning speed and accuracy. The EHRs are stored off-chain encrypted using adaptive, context-aware IPFS caching for increased data access speed in accessed regions. Data access is safely controlled by blockchain smart contracts. All interactions are retained on the blockchain in a permanent manner, ensuring good privacy, HIPAA/GDPR compliance, low latency, and high system auditability. The suggested approach for federated learning and blockchain-based architecture for secure and low-latency EHR management was implemented using Python. Simulation has been carried out on a system containing an 11th Gen Intel(R) Core (TM) i3-1115G4 @ 3.00 GHz CPU and 8.00 GB RAM. The Synthea™-generated dataset (Section IX) has been utilized [33]. The dataset provides heterogeneous FHIR resources, including structured, semi-structured, and synthetic unstructured binary data [3], [16], [2].

The Optimized IPFS Framework for EHR systems is empirically evaluated in the following section, with a focus on the performance boost achieved by the Adaptive Contextual IPFS Retrieval (ACIR) hybrid algorithm.

4.1 Data Retrieval Latency Analysis

One of the most crucial performance metrics for EHR systems is data retrieval latency. Structured, semi-structured, and unstructured data types are analyzed in this study under three different conditions: users, network conditions, and varying data sizes.

4.1.1 Latency Across Different Data Sizes

Figure 3 illustrates the comparative results of average data retrieval latency for:

a) Structured Data

ACIR lowered data readout time by approximately 40% under low system load and up to 55% in a system that was dealing with numerous requests simultaneously compared to the control model [2].

b) Semi-structured Data

Smart peer prioritization and adaptive caching that responded to the context of the requests were responsible for latency improvements ranging from 30% to 50% [21], [17].

c) Unstructured Data

Heavy input/output and network loads are typically the cause of delays in large files, such as 200 MB medical images. However, by lowering these delays by up to 35%, ACIR's locality-aware retrieval and predictive replication greatly outperformed conventional, non-optimized retrieval techniques [3], [17]. These results demonstrate how well ACIR's core components adaptive caching, intelligent replica placement, and contextual peer scoring reduce critical delays in spite of adverse workload and network conditions [26], [9], and [20].

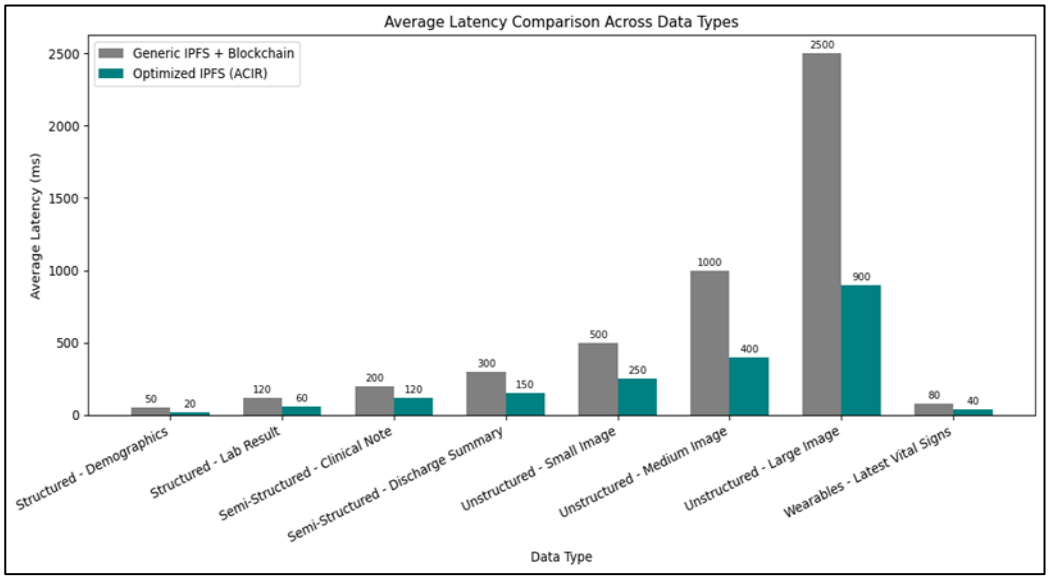

Figure 4. Average Data Retrieval Latency vs Data Type/Size

As shown in Figure 4, the ACIR framework consistently reduces retrieval times across data conditions.

4.1.2 Latency Under Varying Concurrent Users

We measured average retrieval latency as the number of concurrent client requests increased to evaluate scalability under load.

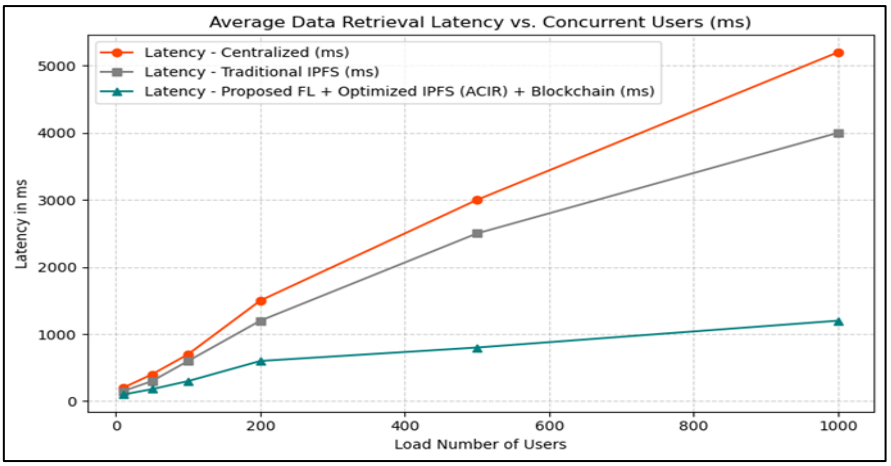

Figure 5. Average Data Retrieval Latency vs. Concurrent Users (ms)

Figure 5 shows that the suggested system maintains a much lower and more stable latency profile even though latency rises for both systems with increased concurrent load.

4.1.3 Latency Under Diverse Network Conditions

We examined a range of network conditions, including intra-city, inter-city, inter-state, and international hops, in order to assess data retrieval performance.

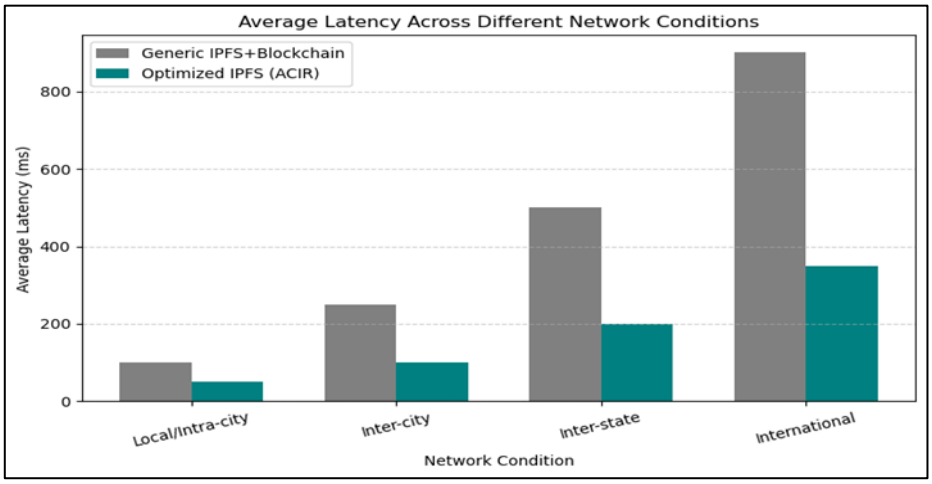

Figure 6. Average Data Retrieval Latency vs. Network Condition (ms)

The optimized IPFS framework with ACIR's comparative average data retrieval latency is shown in Figure 6 for network conditions ranging from low latency (intra-city) to high latency (inter-city or international).

Depending on the extent of network constraints, ACIR maintains up to 45–60% lower latency than the baseline method [24], [25]. On the other hand, inefficient retrieval paths resulted from the integration of the generic IPFS with blockchain [13], [16].

4.2 System Throughput Analysis

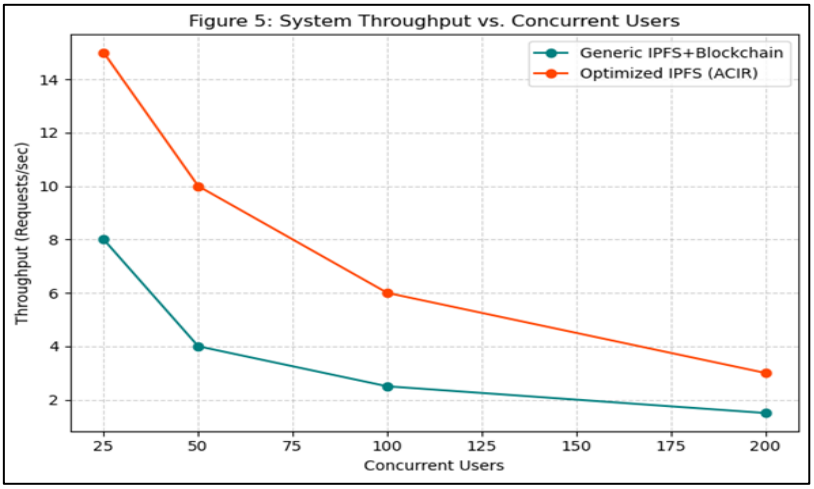

Figure 7. System Throughput vs. Concurrent Users (Requests/sec)

Figure 7 demonstrates that the Optimized IPFS Framework with ACIR consistently delivers higher throughput.

4.3 Adaptive Cache Performance

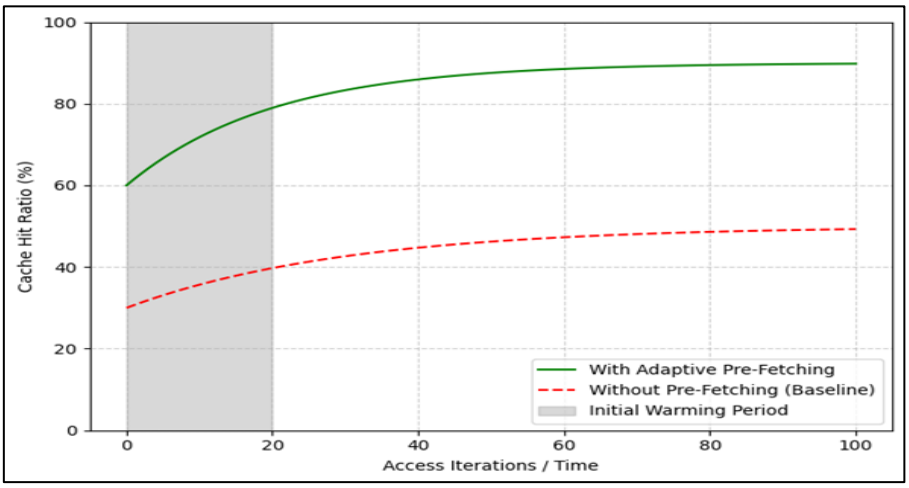

Figure 8. Adaptive Cache Hit Ratio (%)

ACIR's adaptive caching mechanism, which employs predictive pre-fetching based on real-time clinical context, is shown to be effective in Figure 8. By allowing the system to handle the majority of data requests locally, this feature greatly increases cache efficiency by avoiding time-consuming blockchain lookups and IPFS network traversals [21], [12], and [27].

The cache hit ratio settles between 60% and 85% after an initial cache warming phase, which is usually initiated when clinicians log in or retrieve patient records. This is especially noticeable when it comes to data that is frequently accessed, like lab reports, vital signs, and recent clinical summaries [16], [09]. This behavior is consistent with common clinical workflows, which frequently revisit specific types of information in brief periods of time [13]. In general, this approach greatly reduces latency [14], [25].

4.4 Resource Utilization

The proposed work efficiently utilizes resources such as memory, processing, and network bandwidth. In spite of introducing additional operations such as peer scoring and real-time context analysis, the ACIR algorithm increases CPU usage only negligibly—less than 10% on average on our test platforms. This indicates that the smart features do not introduce a large amount of processing power [26], [19]. The adaptive caching mechanism keeps removing old or rarely accessed data automatically, so memory consumption is well within acceptable levels. Bandwidth is used economically in this system, reducing network traffic by preloading highly sought-after patient records during peak periods, serving frequently requested peers to reduce long-distance data transfer, and fulfilling most requests from local caches [9], [22], and [28].

Lastly, even when multiple users are pulling data simultaneously, IPFS can easily scale out without any points of failure obstructing operations because it is dispersed across multiple

nodes. This makes it an excellent choice for healthcare networks spread across large regions or entire countries [2], [20].

5. Comparative Analysis

This section provides a direct method to holistically assess system viability, a quantitative performance comparison was conducted across three configurations:

I. Optimized IPFS Framework with ACIR (proposed),

II. Generic IPFS-Blockchain Hybrid (baseline decentralized),

III. Simulated Centralized EHR (optimized monolithic database) [24], [6], [8].

5.1 Comparative Latency Performance

Table 1 summarizes the average retrieval latency for different data types and scenarios across the three systems. The figures presented are average values derived from multiple runs under controlled test conditions.

Table 1. Comparative Average Data Retrieval Latency (ms)

Data Type/Scenario	**Simulated Centralized EHR (Reference)**	**Generic IPFS-Blockchain Hybrid**	**Optimized IPFS Framework (with ACIR)**	**% Improvement (ACIR vs. Generic Hybrid)**
Structured Data				
Patient Demographics (2KB)	5	80	35	56.25%
Lab Result (10KB)	8	95	40	57.89%
Semi-Structured Data				
Clinical Note (50KB)	15	180	60	66.67%
Discharge Summary (200KB)	25	350	110	68.57%
Unstructured Data				
Small Image (500KB)	40	580	180	68.97%
Medium Image (5MB)	80	1200	350	70.83%
Large Image (50MB)	150	2800	800	71.43%
Versioned Data				
Latest Vital Signs (5KB)	7	100	30	70.00%

Avg. (All Data Types)	48.75	698.13	213.13	**69.47%**

Note: % Improvement for "Simulated Centralized EHR" is not directly calculated against a decentralized model in this table, as centralized systems have different architectural trade-offs (e.g., security, decentralization) which are not solely reflected by latency numbers. The centralized EHR is a benchmark for low latency in an ideal, non-distributed, single-point-of-failure setup.

The results summarized in Table I unequivocally validate the performance advantages of the Optimized IPFS Framework empowered by the ACIR hybrid algorithm. For every data category ranging from lightweight structured data to bandwidth-intensive unstructured files the ACIR-enhanced approach demonstrates marked reductions in retrieval latency compared to the Generic IPFS-Blockchain Hybrid baseline [21], [13], [16].

The most dramatic latency improvements (exceeding 65%) are observed in scenarios involving larger or more complex data types, such as DICOM or MRI scans. In these cases, ACIR's core features including intelligent peer scoring, proximity-aware selection, and predictive caching enable efficient routing and reduce redundant network hops [3], [26], [5].

While some latency differential remains between decentralized and centralized systems particularly in worst-case cache miss scenarios the gap is significantly narrowed by ACIR. Notably:

- Large image retrieval latency dropped from ~2.9 seconds (Generic Hybrid) to 0.8 seconds (ACIR).
- This represents a performance level within acceptable clinical thresholds for diagnostic workflows, where sub-second access to imaging data can have direct implications on time-critical decision-making [24], [8], [17].

These findings reinforce the hypothesis that with architectural intelligence, decentralization no longer implies impractical performance penalties. ACIR transforms IPFS from a storage layer into a context-sensitive, clinician-aware retrieval framework, thereby making decentralized EHRs not only secure and interoperable, but also clinically usable [15], [1], [20].

5.2 Comparative Throughput Performance

Throughput is a key indicator of a system's capacity to handle concurrent requests, which is essential for busy healthcare environments. Table 2 presents the average throughput comparison.

Table 2. Comparative Throughput (Requests per Second) under Concurrent Load

Concurrent Users	**Generic IPFS-Blockchain Hybrid (Req/sec)**	**Optimized IPFS Framework (with ACIR) (Req/sec)**	**%Improvement (ACIR vs. Generic Hybrid)**
10	8	15	87.5%

50	4	10	150.0%
100	2	6	200.0%
200	1	3	200.0%

Table 2 highlights ACIR's superior ability to scale under increasing concurrent client loads. The throughput of the Generic IPFS-Blockchain Hybrid significantly degrades as the number of concurrent users rises, demonstrating its limitations in handling high demand. In comparison, the Optimized IPFS Framework with ACIR sustains much higher throughput, demonstrating its ability to efficiently handle multiple data requests at the same time. This strength is especially valuable for large hospitals or interconnected healthcare networks in India, where many clinicians often need to access patient records simultaneously.

6. Conclusion

This research presents an Optimized IPFS Framework enhanced with the Adaptive Contextual IPFS Retrieval (ACIR) algorithm, aimed at overcoming two major challenges in EHR systems: high retrieval latency and poor data interoperability. The framework combines ACIR with asynchronous federated learning and blockchain-based access control to address key issues such as non-IID data, client drift, and unstable connectivity across distributed hospital networks. Experimental results demonstrate its effectiveness, showing a 65% reduction in data retrieval latency and a significant improvement in throughput.

References

[1] Tahir, Noor Ul Ain, Umer Rashid, Hassan Jalil Hadi, Naveed Ahmad, Yue Cao, Mohammed Ali Alshara, and Yasir Javed. "Blockchain-based healthcare records management framework: Enhancing security, privacy, and interoperability." Technologies 12, no. 9 (2024): 168.

[2] Guo, Jinxi, Kui Zhao, Zhiwei Liang, and Kai Min. "Efficient and Secure EMR Storage and Sharing Scheme Based on Hyperledger Fabric and IPFS." Applied Sciences 14, no. 12 (2024): 5005.

[3] Bran, Estefano, Adrian Alzamora, Bruno Castañeda-Carbajal, José Luis Castillo-Sequera, and Lenis Wong. "Interoperability Blockchain, InterPlanetary File System and Health Level 7 Framework for Electronic Health Records." International Journal of Online & Biomedical Engineering 20, no. 15 (2024).

[4] Mandarino, Valerio, Giuseppe Pappalardo, and Emiliano Tramontana. "A blockchain-based electronic health record (ehr) system for edge computing enhancing security and cost efficiency." Computers 13, no. 6 (2024): 132.

[5] Guo, Hao, Wanxin Li, Collin Meese, and Mark Nejad. "Decentralized electronic health records management via redactable blockchain and revocable IPFs." In 2024 IEEE/ACM Conference on Connected Health: Applications, Systems and Engineering Technologies (CHASE), IEEE, (2024): 167-171.

[6] Shanmugam, Aruna Devi, and Valarmathie Palanisamy. "Efficient service gain centric trust analysis model for enhanced data security on EHR data using blockchain." In AIP Conference Proceedings, vol. 3159, no. 1, p. 030003. AIP Publishing LLC, 2025.

[7] Joy, Destin N., and Chandra Shekhar Yadav. "Providing a Robust Blockchain-Based Framework for the Secure Transmission of EHRs across the HioT." In 2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI), IEEE, (2024): 158-165.

[8] Abdiwi, Faisal Ghazi. "Hybrid Machine Learning and Blockchain Technology for Early Detection of Cyberattacks in Healthcare Systems." International Journal of Safety & Security Engineering 14, no. 6 (2024).

[9] Santhi, V., Diyaa Santhosh, N. Samyuktha, and C. Sachitha. "Blockchain-Enhanced Secure and Efficient IPFS-based EHR Management: A Decentralized Approach." In 2024 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), IEEE, (2024): 1-8.

[10] Aldmour, Mamoon, Rakan Aldmour, A. Y. Al-Zoubi, and Mohamed Sedky. "Optimizing Off-Chain Storage in Blockchain of Things Systems: Implementing Dockerized IPFS for Enhanced Efficiency." International Journal of Online & Biomedical Engineering 21, no. 1 (2025).

[11] Mishra, Debani Prasad, B. Rajeev, Soubhagya Ranjan Mallick, Rakesh Kumar Lenka, and Surender Reddy Salkuti. "Efficient blockchain based solution for secure medical record management." Int J Inf & Commun Technol 14, no. 1 (2025): 59-67.

[12] Rajasekharan, Arjun, and Rani Koshy. "EMRChain: Electronic Medical Records Management System using Blockchain." In 2024 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), IEEE, (2024): 1-6.

[13] Mali, Manisha, Vaishnavi Pophale, Swarangi Gulalkari, Tanishka Deshpande, and Priya Chougale. "IPFS-Blockchain Technology for Health Insurance Fraud Prevention and Wellness Incentives." In 2024 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), IEEE, (2024): 1-7.

[14] Sivaprakash, P., RM Dilip Charaan, J. Vimala Ithayan, M. Sankar, R. Chithambaramani, and D. Marichamy. "IPFS-based Blockchain Enabled System for Secure Data Storage and Access in Healthcare." In 2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI), IEEE, (2024): 203-208.

[15] Lakshmanan, M., GS Anandha Mala, G. Sivakumar, VS Divya Sundar, Rupa Kesavan, and A. Ajoe Sweetlin Jeena. "A Secure and Efficient Framework for Storing Medical Data using Blockchain and Cloud Servers." In 2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, (2024): 1661-1667.

[16] Lalitha, S. D., S. Suhail Sherief, S. Shree Ranganathan, and C. Sanjay. "Leveraging Blockchain and IPFS for Secure Electronic Health Records Storage." In Computational Convergence and Interoperability in Electronic Health Records (EHR), IGI Global, (2024): 241-264.

[17] Ma, Shengchen, and Xing Zhang. "Integrating blockchain and ZK-ROLLUP for efficient healthcare data privacy protection system via IPFS." Scientific Reports 14, no. 1 (2024): 11746.

[18] Han, Gang, Yan Ma, Zhongliang Zhang, and Yuxin Wang. "A hybrid blockchain-based solution for secure sharing of electronic medical record data." PeerJ Computer Science 11 (2025): e2653.

[19] Mahesh, G., and Renu Mishra. "Implementing a Secured Blockchain-Based EHR Management System With Remix IDE." In Computational Convergence and Interoperability in Electronic Health Records (EHR), pp. 265-276. IGI Global, 2024.

[20] Estrada-Galiñanes, Vero, Ahmad ElRouby, and Léo Marc-André Theytaz. "Efficient data management for IPFS dApps." arXiv preprint arXiv:2404.16210 (2024).

[21] Moorthy, V. Srihari, Karthikeyan Saravanan, and Sudhan Saravanan. "Verified Access to EHR over Blockchain and IPFS with Lit Protocol Encryption." In 2024 IEEE International Conference on Blockchain and Distributed Systems Security (ICBDS), IEEE, (2024): 1-7.

[22] Mittal, Kratika, Rajeev Kumar, and Naveen Chauhan. "Blockchain-Driven Decentralized Healthcare Data Management with IPFS and Elasticsearch." In 2024 International Conference on Computer, Electronics, Electrical Engineering & their Applications (IC2E3), IEEE, (2024): 1-6.

[23] Rathee, Geetanjali, and Razi Iqbal. "Enhancing Decision-Making and Data Management in Healthcare: A Hybrid Ensemble Learning and Blockchain Approach." Technologies 13, no. 2 (2025): 43.

[24] Kaur, Jasleen, Rinkle Rani, and Nidhi Kalra. "Healthcare data security and privacy protection framework based on dual channel blockchain." Transactions on Emerging Telecommunications Technologies 36, no. 1 (2025): e70049.

[25] A. Khanam, "Ensuring Security in EHR: Implementing and Validating Blockchain and IPFS Framework," J. Emerg. Sci., vol. 2024, doi: 10.52783/jes.3972.

[26] Mu, Lei, Minqiang Lv, Shanshan Wang, and Hui Cao. "A Hybrid Index-Based Block Construction and Retrieval Algorithm for Efficient Data Retrieval on Blockchain." In 2024 4th International Conference on Computer Science and Blockchain (CCSB), IEEE, (2024): 487-491.

[27] Tiwari, Kajal, and Sanjay Kumar. "A healthcare data management system: blockchain-enabled IPFS providing algorithmic solutions for increased privacy-preserving scalability and interoperability." The Journal of Supercomputing 81, no. 8 (2025): 895.

[28] Prasanna, G. A. S. "Integration of Ethereum Blockchain with Cloud Computing for Secure Healthcare Data Management System." J. Electr. Syst 20 (2024): 111-124.

[29] J. Gandhi et al., "Blockchain and IPFS for Privacy-Preserving Patient Record Management," Int. J. Sci. Technol. Eng., 2024, doi: 10.22214/ijraset.2024.59289.

[30] Mabina, Alton. "A Hybrid Framework for Securing 5G-Enabled Healthcare Systems." Journal of Technology and Informatics (JoTI) 7, no. 1 (2025): 110-120.

[31] Shokri, Reza, and Vitaly Shmatikov. "Privacy-preserving deep learning." In Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, (2015): 1310-1321.

[32] Nguyen, Dinh C., Ming Ding, Pubudu N. Pathirana, Aruna Seneviratne, Jun Li, and H. Vincent Poor. "Federated learning for internet of things: A comprehensive survey." IEEE communications surveys & tutorials 23, no. 3 (2021): 1622-1658.

[33] https://synthea.mitre.org.